## AMENDMENTS to the DRAWINGS

No amendments or changes to the Drawings are proposed.

## REMARKS

**Reconsideration**

We appreciate the Examiner's withdrawal of the rejections under 35 U.S.C. §103(a) over Patel, Sanchez and Stevens.

**Claim Objections**

With respect to the objections regarding the typographical error "attributed", the objection was made to Claims 1, 20, and 23, but Claim 1 was cancelled. We believe this objection was intended to be over Claims 8, 20 and 23. We have amended Claims 8, 20 and 23 to make this read "attribute" as the Examiner believed was the intention.

With respect to the objection to Claim 26 for failing to end with a period, we have corrected this as well.

Reconsideration of the objections is respectfully requested.

**Rejections over 35 U.S.C. §112, First Paragraph**

With respect to the rejections regarding whether or not the Applicant(s) possessed the invention aspects *"responsive to said resolving, converting said obtained attribute value from a first value format to a second value format, wherein said first value format is incompatible with said directory access protocol, and wherein said second value format is compatible with said directory access protocol"*, we respectfully disagree.

Compatibility Conversion. Please note that in our ¶0011 (as numbered in the pre-grant publication), we introduced the history of incompatible and proprietary protocols which were well known in the industry. Then, in ¶0012, we discussed the open standard X.500 for directory services, and that various directory server protocol extension approaches (e.g. "add-ons") have been taken to provide bridging technologies between X.500 and protocols which are not X.500 compatible. Please also see our ¶¶0087 and 0137, in which we describe our own "extension" to the directory server to allow for compatibility between old directory servers and new directory servers employing our invention.

Regarding "converting" data to be LDAP compatible, *per se*, please see our Figure 4, which is described in ¶0057 as illustrating conversion of dynamic data (e.g. time-varying data) to static data for storage in an LDAP directory. We contend that no art of record shows a standard LDAP directory storing anything other than *static* data, but if the Examiner disagrees, we respectfully ask for the Examiner to provide any extrinsic evidence or an affidavit of knowledge (37 C.F.R. §1.104(d)(2)). Otherwise, we believe that dynamic data is inherently incompatible with storage in an LDAP directory.

We believe that the existing technologies merely took a snapshot of a dynamic value, and stored that snapshot value as a static value in the LDAP directory. Please note that this is not the same as what we have claimed – we have claimed avoiding storing of the dynamic value in the LDAP directory because this incurs a long processing delay. Instead, we have claimed converting the non-LDAP formatted data into LDAP-formatted data and inserting it directly into the response to the requester, but avoiding actually storing the converted data in the LDAP directory at all (because it will probably be different the next time it is requested).

LDAP Format. By LDAP format, we are referring to the concepts and characteristics of LDAP-standard databases and access protocol messages as described in ¶¶0015 - 0026. Please note that, per the previous paragraph, applicants state in ¶0026 that LDAP directories store *static* values for attributes. Please consider that this statement is part of the record to which the inventors have sworn to be believed to be true to the best of their knowledge. If the Examiner disagrees or knows of art in which *dynamic* data (e.g. time-varying data) is directly stored in an LDAP directory, we respectfully request that extrinsic evidence or an affidavit to be placed in the examination record to support that holding.

"Responsive to. . . ". Regarding the phrase "*responsive to said resolving, converting said obtained attribute value from a first value format to a second value format, . . .* ", we respectfully point out that the decision block #82 of Figure 8 shows responding to determining that the attribute is "real-time" (e.g. dynamic), then invoking a real-time attribute processor (RTAP). Please note that Figure 5 shows the RTAP getting the current value (#62) of the requested real-time value so that it can be returned to the requester (application 2 #23). Please note that ¶0131 states *". . . the appropriate RTAP (51) process or method, which in turn retrieves or otherwise determines (62) current real-time value of the attribute(s) based upon a dynamic data source (54). These real-time values then are returned (34") to the LDAP server and the requesting client (23) as if it were a normal result from a static attribute return. "*

We respectfully submit that one skilled enough to make the combinations and changes as proposed in the rejections under 35 U.S.C. §103(a) would also be capable of determining that *"as if it were a normal result from a static attribute return"* means that it would be converted (as if) to an LDAP format (normal format for an LDAP request).

Avoiding Storing of Converted Attribute Value in the LDAP Directory. As argued in the previous paragraph, we have specified in our claims that the requested dynamic attribute value is not simply converted and stored in the LDAP directory because this incurs processing time on each request, and for values which are often requested, this bogs down the LDAP server. Instead, we have specified in our claims *". . . while suppressing or avoiding storing of said converted attribute value in said directory structure"*. Please note that this aspect of our invention is especially highlighted in ¶0133-0134 (updating attribute value in the directory is eliminated), stale snapshot stored in the directory are not necessary (¶0136), but the interface to

the LDAP is unchanged (e.g. LDAP compatible, including all attributes in the return message) (¶0137).

We believe, based on the presumed skill level required for the proposed 103 combination, that our disclosure would be sufficient to demonstrate possession of our invention with regards to all aspects of our claimed invention.

If the Examiner disagrees, then we respectfully request an explicit determination of the ordinary skill level at the time of our invention. We believe it would be fundamentally unfair to employ a relatively high ordinary skill level to establish obviousness rejections under 35 U.S.C. §103(a), but to employ a relatively low skill level to establish insufficient disclosure rejections under 35 U.S.C. §112, first paragraph.

**Rejections under 35 U.S.C. §103(a)**

For brevity of the examination record, we respectfully maintain and incorporate herein from our previous reply(ies) our arguments regarding the teachings of previously-cited Patel and Sanchez references.

Regarding the rejections over newly-cited Robb in combination with Sanchez and Patel, we respectfully disagree that Robb teaches converting the real-time value from a first non-compatible format to a second compatible format. It was argued that Robb teaches our claimed conversion from a first format (now recited as a directory access protocol incompatible real-time attribute) to a second format (now recited as a directory access protocol compatible static attribute). Please note that we have amended the claims slightly to clarify that the real-time attribute value is originally incompatible with a directory access protocol (corresponding to a "first format"), and that it is converted to a static attribute compatible with a directory access protocol (corresponding to a "second format").

Robb's ¶0074 appears to discuss work flow management rules and a rules engine in general, but there appears to be no specific disclosure regarding conversion of real-time attributes to a directory access protocol compatible static attribute. Robb's ¶0076 appears to discuss applications containers and services in general, including unified messaging of some sort, but there is also no mention of converting a real-time LDAP-incompatible attribute to a static LDAP-compatible attribute.

We have searched the entirety of the Robb disclosure, and can only one reference to "voice compatible" applications (¶0073), but there appears to be no other reference to compatibility at all. Similarly, we have searched but found no references to real-time attributes anywhere in Robb's disclosure.

Perhaps the Examiner's argument rests of Robb's disclosure of "*aggregating* static and

dynamic content" (¶0076) by their "Content Manager". This appears in their Claim 32, as well, but the term "static" does not appear anywhere but in ¶0076 and Claim 32. The term "aggregate" appears throughout Robb's disclosure in reference to aggregating services, but still there appears to be no detail regarding 'aggregating' static and dynamic data into an LDAP return message, else there would need to be some sort of disclosure regarding how a dynamic attribute can be represented in a protocol message which is only defined to carry static values retrieved from a database of static values.

For a better or clearer definition of what is meant by Robb by "aggregating", we ask the Examiner to consider the first recitation of the term in the disclosure at ¶0008. Here, "aggregating" is referring some action performed on an Application Service Provider (ASP), who are business entities (¶0004). We believe that Robb's use of the term "aggregate" refers to a business entity and the underlying technologies for that business model to bundle and license online services (¶0086).

With respect to "aggregation" of data in this context, such as Robb's Claim 24 which "aggregates" data for presentation to end user devices, it is not clear or stated that "end user devices" include LDAP directory clients. Robb's ¶0013 appears to refer to a technical form aggregation performed by their AIP (their platform) which is a service bundling or packaging platform for subscription purposes. But, we are unable to find any disclosure that shows that their aggregation converts LDAP-incompatible real-time values to LDAP-compatible static attributes in response to an LDAP access or read request, or that their aggregator returns such a converted value directly in the LDAP return message without storing it in the LDAP directory, as we have claimed.

For these reasons, we respectfully request reconsideration of these rejections, especially in view of the particular amendments made herein.

**Ordinary Skill Level**

We respectfully repeat our request for an explicit determination by the Examiner of what is being considered to have been the ordinary skill level in the relevant art(s) at the time of our invention, and that analysis be placed into the record of examination. Such a determination is relevant to the rejections under 35 U.S.C. §112 as well as under 35 U.S.C. §103(a).

**Request for Indication of Allowable Subject Matter**

We believe we have responded to all grounds of rejection and objection, but if the Examiner disagrees, we would appreciate the opportunity to supplement our reply.

We believe the present amendment places the claims in condition for allowance. If, for any reason, it is believed that the claims are not in a condition for allowance, we respectfully request constructive recommendations per MPEP 707.07(j) II which would place the claims in condition for allowance without need for further proceedings. We will respond promptly to any Examiner-initiated interviews or to consider any proposed examiner amendments.

Respectfully,

/ Robert Frantz /

Robert H. Frantz, Reg. No. 42,553
Agent for Applicant
Tel: (405) 812-5613
Franklin Gray Patents, LLC